Ex No:
Date:

SIMPLE INTEREST

AIM:

To write a C Program to calculate Simple Interest.

ALGORITHM:

```
STEP 1: Start the program.
STEP 2: Input the values of Principle amount(p), number of Years(n), rate of Interest(r).
STEP 3: Calculate the simple interest using the formula(si)
                              si = (p*n*r)/100
STEP 4: Print the calculated value as Simple Interest.
STEP 5: Stop the program.
```

PROGRAM FOR SIMPLE INTEREST CALCULATION

```
#include<stdio.h>
#include<conio.h>
void main()
{
float p,n,r;
float si;
clrscr();
printf("Enter the value of Principle Amount :\n");
scanf("%f",&p);
printf("Enter the number of Years :\n");
scanf("%f",&n);
printf("Enter the rate of Interest :\n");
scanf("%f",&r);
si = (p*n*r)/100;
printf("Simple Interest value is :%f",si);
getch();
}
```

OUTPUT:

```
Enter the value of Principle Amount : 5000
Enter the number of Years : 3
Enter the rate of Interest : 9
Simple Interest value is : 1350.000000
```

RESULT:

Ex No:
Date:

REVERSE A NUMBER

AIM:

To write a C Program to reverse five digit numbers.

ALGORITHM:

```
STEP 1: Start the program
STEP 2: Initialize reverse=0.
STEP 3: Read digit
STEP 4: Check whether digit>0 then go to step 5
            else go to step 9
STEP 5: reverse=reverse*10
STEP 6: reverse=reverse+digit%10
STEP 7: digit=digit/10
STEP 8: Go to step 4
STEP 9: Print reverse
STEP 10: Stop
```

PROGRAM TO REVERSE A NUMBER

```
#include<stdio.h>
#include<conio.h>
void main()
{
int reverse=0,num,rem;
clrscr();
printf("Enter the number:\n");
scanf("%d",&num);
while(num>0)
{
rem=num%10;
reverse=(reverse*10)+rem;
num=num/10;
}
printf("Reverse number=%d\n",reverse);
getch();
}
```

OUTPUT:

```
Enter the number: 7539
Reverse number= 9357
```

RESULT:

Ex No:
Date:

INCOME TAX CALCULATION USING NESTED IF

AIM:

To write a C Program to calculate Income tax using nested If.

ALGORITHM:

```
STEP 1: Start the program.
STEP 2: Read the value of income.
STEP 3: Check if value<=22000
            Yes: Assign tax=0
            No:  Go to step 4
STEP 4: Check if value<=30000
            Yes: Assign tax=(income-22000)*0.20
            No:  Go to step 5
STEP 5: Check if value<=50000
            Yes: Assign tax=(income-30000)*0.30
            No:  Go to step 6
STEP 6: Check if value<=100000
            Yes: Assign tax=(income-50000)*0.40
            No:  Go to step 7
STEP 7: else
            Assign tax=(income-100000)*0.50
STEP 8: Print the value of income
STEP 9: Stop
```

PROGRAM FOR NESTED IF

```
#include<stdio.h>
#include<conio.h>
main()
{
float income,tax;
printf("Enter the basic income:\n");
scanf("%f",&income);
if(income<=22000)
tax=0;
else if(income<=30000)
tax=(income-22000)*0.20;
else if(income<=50000)
tax=(income-30000)*0.30;
else if(income<=100000)
tax=(income-50000)*0.40;
else
tax=(income-100000)*0.50;
printf("Tax amount %8.2f",tax);
getch();
}
```

OUTPUT:

```
Enter the basic income: 35000
Tax amount: 1500
```

RESULT:

Ex No:
Date:

PRIME NUMBER CHECKING

AIM:

To write a C Program to check prime numbers.

ALGORITHM:

```
STEP 1: Start the program
STEP 2: Read the values of n,c.
STEP 3: Initialize c=2
STEP 4: if(n%c==0)
            Yes: %d is prime
            No: %d is not prime
STEP 5: check if it is prime or not prime
STEP 6: Stop the program
```

PRIME NUMBER CHECKING

```
#include<stdio.h>
#include<conio.h>
main()
{
int n,c;
printf("Enter a number to check if it is prime\n");
scanf("%d",&n);
for(c=2;c<=n-1;c++)
{
if(n%c==0)
{
printf("%d is not prime \n",n);
break;
}
}
if(c == n)
{
printf("%d is prime \n",n);
}
getch();
}
```

OUTPUT:

```
Enter a number to check if it is prime 11
11 is prime
```

RESULT:

Ex No:
Date:

SUM OF DIGITS

AIM:

To write a C Program for calculate sum of digits.

ALGORITHM:

```
STEP 1: Start the program.
STEP 2: Read the value of num.
STEP 3: Repeat the step 4 and 5 until num>0
STEP 4: Calculate
            rem=num%10;
            sum=sum+rem;
            num=num/10;
STEP 5: Print the sum value
STEP 6: Stop the program
```

PROGRAM FOR SUM OF DIGITS

```
#include<stdio.h>
#include<conio.h>
void main()
{
int rem,sum=0,num;
clrscr();
printf("Enter any number:\n");
scanf("%d",&num);
while(num>0)
{
rem=num%10;
sum=sum+rem;
num=num/10;
}
printf("The sum of digits is %d",sum);
getch();
}
```

OUTPUT:

```
Enter any number: 123
The sum of digits is 6
```

RESULT:

Ex No:
Date:

QUADRATIC EQUATION

AIM:

To write a C Program to find the roots of a Quadratic equation.

ALGORITHM:

```
STEP 1:Start.
STEP 2:Read a, b, c values.
STEP 3: Compute d = (b*b)-(4*a*c)
STEP 4: if d > 0 then, x = b+sqrt (d)/(2*a) ,y = b-sqrt(d)/(2*a)
STEP 5: Otherwise if d = 0 then, compute x = -b/2a, y = -b/2a.
STEP 6: print x and y values.
STEP 6: Otherwise if d < 0 then print roots are imaginary.
STEP 7:  Stop
```

PROGRAM FOR QUADRATIC EQUATION

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
void main()
{
int a,b,c;
float d,x,y;
printf("\n\t\t\t quadratic equation\n");
printf("\n\t\t\tEnter the value of a:");
scanf("%d",&a);
printf("\n\t\t\tEnter the value of b:");
scanf("%d",&b);
printf("\n\t\t\t Enter the value of c:");
scanf("%d",&c);
d=(b*b)-(4*a*c);
if(d>0)
{
x=(-b+sqrt(d))/(2*a);
y=(-b-sqrt(d))/(2*a);
printf("\n\t\t\t roots are real and unequal");
printf("\n\t\t\t value of x:%f",x);
printf("\n\t\t\t value of y:%f",y);
}
else if(d==0)
{
x=(-b)/(2*a);
y=(-b)/(2*a);
printf("\n\t\t\t Roots are real and equal");
printf("\n\t\t\t value of x:%f",x);
printf("\n\t\t\t value of y:%f",y);
}
else
{
x=(-b)/(2*a);
y=(sqrt(-d))/(2*a);
printf("\n\t\t\t Roots are imaginary");
printf("\n\t\t\t value of x:%f",x);
printf("\n\t\t\t value of y:%f",y);
}
getch();
}
```

OUTPUT:

```
Enter the value of a: 1
Enter the value of b: 2
Enter the value of c: 1

Roots are real and equal
value of x: -1.000000
value of y: -1.000000
```

RESULT:

Ex No:
Date:

SINE SERIES

AIM:

To write a C Program to find sine series.

ALGORITHM:

```
STEP 1: Start
STEP 2: x,n,i,j,s,f,k;
STEP 3: Enter x value
STEP 4: Read x
STEP 5: Repeat steps 6 to 11 for(i=1.0 to i<=n)
STEP 6: Set j=1.0
STEP 7: Repeat steps 8 while(j<=i)
STEP 8: Set f=f*j;j=j+1;
STEP 9: Repeat steps 10 while(k<=i)
STEP 10: Set k=k*(-1.0);
STEP 11: set s=s+(pow(x,i)/f)*k ;
STEP 12: Display sine series
STEP 13: Stop
```

PROGRAM FOR SINE SERIES

```
#include<stdio.h>
#include<math.h>
#include<conio.h>
main()
{
float x,n,i,j,k,f,s;
clrscr();
printf("\n\t\t\t Sum of sine series");
printf("\n\t\t\t Enter x value:");
scanf("%f",&x);
printf("\n\t\t\t Enter n value:");
scanf("%f",&n);
s=0.0;
f=1.0;
k=1.0;
for(i=1.0;i<=n;i=i+2)
{
for(j=1.0;j<=i;j++)
{
f=f*j;
}
s=s+(pow(x,i)/f)*k;
k=k*(-1.0);
}
printf("\n\t\t\t value of x= %f",x);
printf("\n\t\t\t value of y=%f",n);
printf("\n\t\t\t Sine series=%f",s);
getch();
}
```

OUTPUT:

```
Sum of sine series
Enter x value: 2
Enter n value: 3

value of x=2.000000
value of y=3.000000
Sine series=0.666667
```

RESULT:

COSINE SERIES

AIM:

To write a C Program to find cosine series.

ALGORITHM:

```
STEP 1: Start
STEP 2: x,n,i,j,s=0.0,f=1.0,k=1.0;
STEP 3: Enter x value
STEP 4: Read x
STEP 5: Repeat steps 6 to 11 for(i=1.0 to i<=n)
STEP 6: Set j=1.0
STEP 7: Repeat steps 8 while(j<=i)
STEP 8: Set f=f*j;j=j+1;
STEP 9: Repeat steps 10 while(k<=i)
STEP 10: Set k=k*(-1.0);
STEP 11: set s=s+(pow(x,i)/f)*k ;
STEP 12: Display cosine series
STEP 13: Stop
```

PROGRAM FOR COSINE SERIES

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
void main()
{
float x,n,i,j,k=1.0,f=1.0,s=0.0;
clrscr();
printf("\n\t\t\t Sum of cosine series");
printf("\n\t\t\t Enter x value:");
scanf("%f",&x);
printf("\n\t\t\t Enter n value:");
scanf("%f",&n);
s=0.0;
f=1.0;
k=1.0;
for(i=0;i<=n;i=i+2)
{
for(j=1.0;j<=1;j++)
{
f=f*j;
}
s=s+(pow(x,i)/f)*k;
k=k*(-1);
}
printf("\n\t\t\t value of x=%f",x);
printf("\n\t\t\t value of y=%f",n);
printf("\n\t\t\t value of cosine series=%f",s);
getch();
}
```

OUTPUT:

```
Sum of cosine series
Enter x value: 1
Enter n value: 4

value of x: 1.000000
value of y: 4.000000
value of cosine series=1.000000
```

RESULT:

Ex. No:
Date :

EXPONENTIAL SERIES

AIM:

To write a C program for Exponential Series.

ALGORITHM:

```
Step 1: Start the program
Step 2: Read x
Step 3: Read n
Step 4: Initializing i=1.0,j=1.0
Step 5: Testing i<=n, j<=i
Step 6: Increment i++, j++
Step 7: f=f*j
Step 8: print s
Step 9: Stop the program
```

PROGRAM FOR EXPONENTIAL SERIES

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
void main()
{
float x,n,i,j,k,f,s;
clrscr();
printf("\n\t\t\t Sum of exponential series");
printf("\n\t\t\t Enter x value");
scanf("%f",&x);
printf("\n\t\t\t Enter n value");
scanf("%f",&n);
s=1.0;
f=1.0;
for(i=1.0;i<=n;i++)
{
for(j=1.0;j<=i;j++)
{
```

```
f=f*j;
}
s=s+(pow(x,i)/f);
}
printf("\n\t\t\t value of x=%f",x);
printf("\n\t\t\t value of y=%f",n);
printf("\n\t\t\t Exponential series=%f",s);
getch();
}
```

OUTPUT:

```
Enter x value 1
Enter n value 2
value of x=1.000000
value of y=2.000000
Exponential series=2.500000
```

RESULT:

Ex.No:
Date :

PALINDROME CHECKING

AIM:

To wriite a C program for palindrome checking using string functions.

ALGORITHM:

```
Step 1: Start the program
Step 2: Read str1
Step 3: Assign str2 ← str1
Step 4: Check if strcmp (str1, str2)==0
(a) Yes: print given string is palindrome
(b) No : print given string is not palindrome
Step 5: Stop the program
```

PROGRAM FOR PALINDROME CHECKING

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char str1[20],str2[20];
clrscr();
printf("\n\t\t\t Enter the string:");
scanf("%s",&str1);
strcpy(str2,str1);
strrev(str1);
if(strcmp(str1,str2)==0)
printf("\n\t\t\t\t %s is a palindrome",str1);
else
printf("\n\t\t\t\t %is not a palindrome",str2);
getch();
}
```

OUTPUT:

```
Enter the string: madam
madam is a palindrome
```

RESULT:

Ex.No:
Date :

ASCENDING ORDER

AIM:

To write a C program for ascending order of numbers.

ALGORITHM:

Step 1: Start the program
Step 2: Read the value of n
Step 3: Read the value of a[i] using the loop
Step 4: Repeat the Step 5 and 6 for I varying from 0 to n
Step 5: Repeat the Step 6 for j varying from i+1 to n
Step 6: Check if a[i]>a[j]
(a) Yes: Assign t ← a[i],a[j] ← a[i],a[j] ← t
(b) No : go to Step 5
Step 7: print the value of a[i]
Step 8: Stop the program

ASCENDING ORDER

```
#include<stdio.h>
#include<conio.h>
void main();
{
int I,j,a[5],t;
clrscr();
printf(“Sorting of numbers”);
printf(“\n\n Enter 5 numbers”);
for(i=0;i<5;i++)
{
scanf(“%d”,&a[i]);
}
for(i=0;i<5;i++)
{
```

```
for(j=i+1;j<5;j++)
{
if(a[i]>a[j])
{
t=a[i]
a[i]=a[j];
a[j]=t;
}
}
}
printf(“\n Ascending order);
for(j=0;j<5;j++)
{
printf(“\n%d”,a[j]);
getch();
}
}
```

OUTPUT:

```
Sorting of numbers
Enter 5 numbers
56
34
23
11
8
Ascending order
8
11
23
34
56
```

RESULT:

Ex No:
Date:

DESCENDING ORDER OF NUMBER

AIM:

To write a C program for descending order of numbers.

ALGORITHM:

Step 1: Start the program
Step 2: Read the value of n
Step 3: Read the value of number [i]using for loop
Step 4: Repeat the Step 5 & 6 for i varrying from 0 to n
Step 5: Read the Step 6 for j varrying from i+1 to n
Step 6: Check if numbers a[i]<a[j]
(a) Yes: Assign t ←a[i],a[j] ← a[j],a[i] ←t
(b) No : go to Step 5
Step 7: print the value of number[i]
Step 8: Stop the program

DESCENDING ORDER

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,j,a[5],t;
clrscr();
printf(“sorting of numbers”);
printf(“\n\n Enter 5 numbers”);
printf(“\n Enter numbers”);
for(i=0;i<5;i++)
{
```

```
scanf("%d", &a[i]);
}
for(i=0;i<5;i++)
{
for(j=i+1;j<5;j++)
{
if(a[i]<a[j])
{
t=a[i];
a[i]=a[j];
a[j]=t;
}
}
}
printf("\n Descending order");
for(i=0;i<5;i++)
{
printf("\n%d",a[i]);
getch();
}
}
```

OUTPUT:

```
sorting of numbers
Enter 5 numbers
56
18
23
2
8
Descending order
56
23
18
8
2
```

RESULT:

Ex No:
Date:

MATRIX ADDITION

AIM:

To write a C program for matrix addition using function.

ALGORITHM:

```
Step 1: Start the program
Step 2: Read row m and column n value for matrix I & II
Step 3: Read the values for Ist matrix a[i][j]
Step 4: Read the values for Iist matrid b[i][j]
Step 5: Calculate matrix addition
c[i][j]=a[i][j]+b[i][j] using for loop
Step 6: print the added matrix c[i][j]
Step 7: Stop the program
```

MATRIX ADDITION

```
#include<stdio.h>
#include<conio.h>
void main()
{
int m,n,i,j,a[10][10],b[10][10],c[10][10]
clrscr();
printf(“Enter row value for 1 matrix”);
scanf(“%d”,&m);
printf(“Enter column value for 1 matrix”);
scanf(“%d”,&n);
printf(“Enter the value of 1 matrix”);
for(i=-;i<m;i++)
{
for(j=0;j<n;j++)
{
scanf(“%d”,&a[i][j]);
}
}
printf("Enter value of 2 matrix");
for(i=0;i<m;i++)
{
for(j=o;j<n,j++)
{
scanf("%d", &b[i][j]);
}
}
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
c[i][j]=a[i][j]+b[i][j];
}
}
printf("Matrix Addition \n");for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("\t%d",c[i][j]);
}
printf("\n");
}
getch();
}
```

OUTPUT:

```
Enter row value for 1 matrix 2
Enter column value for 1 matrix 2
Enter value of 1 matrix2
2
2
2
Enter value of 2 matrix2
2
2
2
Matrix Addition
4 4
4 4
```

RESULT:

Ex no:

MATRIX SUBTRACTION

Date:

AIM:

To write the C program for matrix subtraction using arrays.

ALGORITHM:

Step 1: Start the program
Step 2: Read row n & column n value for matrix I & II
Step 3: Read the value for Ist matrix a[i][j]
Step 4: Read the value for IInd matrix b[i][j]
Step 5: Calculate matrix subtracted
d[i][j]=a[i][j]-b[i][j]
Step 6: print the subtracted matrix d[i][j]
step 7: Stop the program.

MATRIX SUBTRACTION

```
#include<stdio.h>
#include<conio.h>
void main()
{
int m,n,i,j,a[10][10],b[10][10],d[10][10];
clrscr();
printf("Enter row value for 1 matrix");
scanf("%d",&m);
printf("Enter the column value of 1 matrix");
scanf("%d",&n);
printf("Enter value of 1 matrix");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
scanf("%d",&a[i][j]);
}
}
printf("Enter value of 2 matrix");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
scanf("%d",&b[i][j])
}
}
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
d[i][j]=a[i][j]-b[i][j];
}
}
printf("Subtracted matrix \n");
for(i=0;i<m;i++){
for(j=0;j<n;j++)
{
printf("\t%d", d[i][j]);
}
printf("\n");
}
getch()'
}
```

OUTPUT:

```
Enter the row value for 1 matrix 2
Enter the column vlaue for 1 matrix 2
Enter value of 1 matrix
2
2
2
2
Enter value of 2 matrix
2
2
2
2
Subtracted matrix
0 0
0 0
```

RESULT:

Ex No:

MATRIX MULTIPLICATON

Date:

AIM:

To write C program for matrix multiplication using arrays.

ALGORITHM:

```
Step 1: Start the program
Step 2: Read row m & columnn n value for matrix I & II
Step 3: Read the value for Ist matrix a[i][j]
Step 4: Read the value for Iist matrix b[i][j]
Step 5: Calculate matrix multiplication
e[i][j]=e[i][j]+a[i][j]*b[*][j]
Step 6: print the multiplied matrix e[i][j]
Step 7: Stop the program.
```

MATRIX MULTIPICATON

```
#include<stdio.h>
#include<conio.h>
void main()
{
int m,n,k,a[10][10],b[10][10],e[10][10];
clrscr();
printf(“Enter row value for 1 matrix”)
scanf(“%d”, &m);
pritnf(“Enter column value for 1 matrix”);
scanf(“%d”, &n);
printf(“Enter value of 1 matrix”);
for(i=0;i,m;i++)
{
for (j=0;j<n;j++)
{
scanf(“%d”, &a[i][j]);
}
}
printf(“Enter value of 2 matrix”);
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
scanf(“%d”, &b[i][j]);
}
}
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
e[i][j]=0;
for(k=0;k<m;k++)
{
e[i][j]=e[i][j]+a[i][j]*b[i][j];}
}
}
Printf(“Multiplied matrix \n”);
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf(“\t%d”,e[i][j]);
}
printf(“\n”);
}
getch();
}
```

OUTPUT:

```
Enter row value for 1 matrix 2
Enter the column value for 1 matrix 2
Enter value of 1 matrix 2
2
2
2
2
Enter value fo 2 matrix 2
2
2
2
2
Multiplied matrix
8 8
8 8
```

RESULT:

Ex No:
Date:

FACTORIAL NUMBER USING RECURSIVE FUNCTION

AIM:

To write C program to find factorial using recursion.

ALGORITHM:

Step 1: Start the program
Step 2: Read the value of n
Step 3: Call fact () to find the fuction factorial using recursion
Step 4: print the factorial value written by the fact
Step 5: Stop the function.

FACTORIAL USING RECURSIVE FUNCTION

```
#include<stdio.h>
#include<conio.h>
#include<math.h>
int fact(int n);
void main()
{
int a,n;
clrscr();
printf("Finding factorial \n");
printf("Enter x value:");
scanf("%d",&n);
a=fact(n);
printf("\n\t Factorial value: %d\n",a);
getch();
}
int fact(int n)
{
if(n==1)
{
return(1);
}
else
{
return(n*fact(n-1));
}
}
```

OUTPUT:

```
Finding Factorial
Enter x value: 3
Factorial Value: 6
```

RESULT:

Ex No:
Date:

STRING LENGTH

AIM:

To write C program for finding string length.

ALGORITHM:

```
Step 1: Start the program
Step 2: Initialize count value to zero
Step 3: Check the condition for s[count] not equal to zero than increment count value
Step 4: Allocate memory for character string
Step 5: Get the value for string
Step 6: Find the length of string and print string value
Step 7: Stop the value.
```

STRING LENGTH WITHOUT USING STRLEN();

```
#include<stdio.h>
#include<conio.h>
int len(char*s)
{
int count=0;
while(s[count]!=’\0’)
{
++count;
}
return count;
}
void main()
{
char*str=(char)malloc(1000*size of(char));
clrscr();
printf(“Enter the string \n”);
gets(str);
pritnf(“\n Lenght of the string %d”,len(str));
getch();
}
```

OUTPUT:

```
Enter the string welcome
Lenght of the string
```

RESULT:

Ex No:
Date:

STRING COMPARISON

AIM:

To write a C program for finding string comparison.

ALGORITHM:

Step 1: Start the program
Step 2: Declare the variable , f, s and result
Step 3: Get the value for first string and second string
Step 4: Compare both values, if the result is equal to zero, print as strings are equal, else string not equal
Step 5: Stop the program.

PROGRAM FOR STRING COMPARISON

```
#include<stdio.h>
#include<conio.h>
int compare(char*,char*);
void main()
{
char f[1000];
char s[100], result;
clrscr();
printf("Enter the first string:\n");
gets(f);
printf("Enter the second string:\n");
gets(s);
result=compare(f,s);
if(result==0)
{
printf("\n Strings are same");
}
else
{
printf("\n Strings are not same");
}
getch();
}
int compare(char*f, char*s);
{
while(*f==s*)
{
if(*f=='\0'||*s=='\0')
break;
f++;
s++;
}
if(*f==’\0’&& *s==’/0’)
return 0:
else
return 1;
}
```

OUTPUT:

```
Enter the first string
welcome
Enter the second string
welcome
Strings are same
```

RESULT:

Ex No:
Date:

STRING COPY

AIM:

To write a C program for finding string copy.

ALGORITHM:

Step 1: Start the program
Step 2: Declare the variables f, s and result
Step 3: Get the value of first string to copy
Step 4: Copy the value of first string to another string
Step 5: Stop the program.

PROGRAM FOR STRING COPY

```
#include<stdio.h>
#include<conio.h>
void copy(char*,char*);
void main()
{
char s[100];
char t[100];
clrscr();
printf("Enter the string to copy:\n");
gets(s);
copy(t,s);
printf("Copied string:%s",t);
getch();
}
void copy(char*t,char*s)
{
while(*s)
{
*t=*s;
s++;
t++;
}
*t='\0';
getch90;
}
```

OUTPUT:

```
Enter the string to copy: welcome
copied string: welcome
```

RESULT:

Ex No:
Date:

STRING CONCATENATE

AIM:

Ex No:
Date:

STRING CONCATENATE

AIM:

To write a C program for finding string concatenation.

ALGORITHM:

Step 1: Start the program
Step 2: Initialize two strings
Step 3: Assign length to zero
Step 4: Check the condition s1[lenght] not equal to zero, then increment the value of length
Step 5: Repeat the following steps for j=0 and check the condition for s2[j] not equal to zero
Step 6: Assign the value of string 2 or string 1
Step 7: print the value of string 1 after concatenation
Step 8: Stop the program.

STRING CONCATENATION

```
#include<stdio.h>
#include<conio.h>
void main()
{
char s1[100] = “Welcome to” , s2[] = “CS Department”;
int length,j;
clrscr();
length = 0;
while (s1[lenght]!=’\0’)
{
++length;
}
for(j = 0;s2[j]=’\0’;++j, ++lenght)
{
s1[lenght] = s2[j];
}
s1[lenght] = ‘\0’;
printf(“After concatenation:”);
puts(s1);
getch();
}
```

OUTPUT:

Welcome to CS Department

RESULT:

Ex No:
Date:

STUDENT DETAILS USING STRUCTURES

AIM:

To write a C program for student details using structures.

ALGORITHM:

Step 1: Start the program
Step 2: Assign eng,tam and major mark
Step 3: Assign 3 students details
Step 4: Print the 3 students eng and tam and major marks with correct order
Step 5: Calculate the total marks
Step 6: Assign total marks → s[i]tam+s[i]eng+s[i]major
Step 7: Print the total marks
Step 8: Stop the program

PROGRAM FOR STUDENTS DETAIL USING STRUCTURES

```
#include<stdio.h>
#include<conio.h>
struct student
{
int eng;
int tam;
int major;
};
void main()
{
struct student s[3];
int i,total=0;
clrscr();
for(i=0;i<=2;i++)
{
printf("Enter marks for 3 students\n");
printf("Enter Eng marks\n");
scanf("%d",&s[i].eng);
printf("Enter Tam marks\n");
scanf("%d",&s[i].tam);
printf("Enter Major marks\n");
scanf("%d",&s[i].major);
total=s[i].eng+s[i].tam+s[i]major;
printf("Total marks to s[%d]student=%d",i,total);
getch();
}
}
```

OUTPUT:

```
Enter marks for 3 students:
Enter Eng marks 56
Enter Tam marks 67
Enter Major marks 95Total marks to s[0] students=218
Enter marks for 3 students:
```

```
Enter Eng marks 78
Enter Tam marks 98
Enter Major marks 90
Total marks to s[1] students=266
Enter marks for 3 students:
Enter Eng marks 98
Enter Tam marks 87
Enter Major marks 67
Total marks to s[2] students=25
```

RESULT: